18 citations found. Retrieving documents...
K. D. Cooper, M. Hall, and L. Torczon. *Unexpected side effects of inline substitution - a case study.* Letters on Programming Languages and Systems, 1(1), Mar. 1992.

This paper is cited in the following contexts:

A Comparative Study of Static and Dynamic Heuristics for .. - Arnold, Fink, Sarkar.. (2000) (6 citations) (Correct)

....suffers with inlining limits of 25 and greater. **As** noted earlier, register pressure or cache behavior might account for this effect. **In** addition, in the presence of dynamic dispatch, we must guard virtual calls to preserve safety. **Other unexpected compiler interactions may contribute (see, e.g. [8]) Somewhat surprisingly, a dramatic increase in code size, tripling code size with a 200 expansion limit, Dynamo 00 submission Page 12 Privileged material please do not distribute did not have a dramatic impact on performance.** In the worst case, jess had a slow down of 21.4 when ....

Keith D. Cooper, Mary W. Hall, and Linda Torczon. *Unexpected side effects of inline substitution: A case study.* ACM LOPLAS, 1(1):22–32, March 1992.

Method Inlining in the Titanium Compiler - Cs Semester Project (Correct)

....object creation, but is unlikely to help much with the intraprocedural optimization of surrounding code since the possibility of a call to an external function still exists. **Nevertheless**, further exploration may be warranted for object allocations within inner loops. **Cooper, Hall and Torczon [7] report an interesting case where performance of a numerically intensive Fortran benchmark was adversely affected by an increase in the number of floating point stalls.** The explanation turned out to be related to Fortran s parameter aliasing rules, which forbid any overlap between arrays passed as ....

Cooper, Keith D., Hall, Mary W. and Torczon, Linda. "*Unexpected Side Effects of Inline Substitution: A Case Study*", ACM Letters on Programming Languages and Systems (March 1992).

A Comparative Study of Static and Profile-Based.. - Arnold, Fink, Sarkar, .. (2000) (2 citations) (Correct)

....suffers with inlining limits of 25 and greater. **As** noted earlier, register pressure or cache behavior might account for this effect. **In** addition, in the presence of dynamic dispatch, we must guard virtual calls to preserve safety. **Other unexpected compiler interactions may contribute (see, e.g. [8]) Somewhat surprisingly, a dramatic increase in code size, tripling code size with a 200 expansion limit, did not have a dramatic impact on performance.** In the worst case, jess had a slow down of 21.4 when increasing code expansion from 25 to 200 for DCG E. **We** conclude that for these ....

K. D. Cooper, M. W. Hall, and L. Torczon. *Unexpected side effects of inline substitution: A case study.* ACM LOPLAS, 1(1):22–32, March 1992.

Interprocedural Analysis to Support Static Scheduling of.. - Trung Nguyen Zhiyuan (Correct)

....Foo. **In** this example, A and B map to X and Y respectively at the first call statement and to C and D respectively at the second call statement. **Hence**, the memory location represented by a formal parameter may vary from one invocation of the subroutine to another. **Subroutine in lining [CHT91, CHT92, Hol91] can eliminate all formal parameters, and subroutine cloning [CHK92] can result in a unique virtual address for every formal parameter.** However, in lining and cloning may make the resulting object code overly large and, thus, are performed to limited extent in practice. **When** we determine ....

K. Cooper, M. Hall, and L. Torczon. *Unexpected side effects of inline substitution - a case study.* Letters on Programming Languages and Systems, 1(1), March 1992.

alto: A Platform for Object Code Modification - Muth (1999) (8 citations) (Correct)

....inlining. **ffl** Inlining at the source level might increase the register pressure [26] and lead to suboptimal register allocations. **ffl In FORTRAN programs, the compiler might no be able to exploit the no alias requirement for arguments of subroutine calls, once such a subroutine has been inlined [20].** ffl Inlining of recursive subroutines may lead to stack explosion [14] ffl Through the increase in code size inlining might hurt instruction

cache performance [53] Our optimizer inlines subroutines at the object code level. **This** avoids the problems with the increased register pressure and ....

Keith D. Cooper, Mary W. Hall, and Linda Torczon. *Unexpected side effects of inline substitution: a case study.* ACM Letters on Programming Languages and Systems, 1(1):22–32, March 1992.

Secrets of the Glasgow Haskell Compiler inliner - Jones, Marlow (1999) (14 citations) (Correct)

....we were surprised by the rather small performance increases as the threshold is increased beyond 2, and plan to investigate this further. **8 Related work There is a modest literature on inlining applied to imperative programming languages, such as C and FORTRAN some recent examples are [DH92, CMCH92, CHT91, CHT92].** In these works the focus is exclusively on procedures defined at the top level. **The** benefits are found to be fairly modest (in the 10 20 range) but the cost in terms of code bloat is also very modest. **Considerable** attention is paid to the effect on register allocation of larger basic blocks, ...

K. Cooper, M. Hall, and L. Torczon. *Unexpected Side Effects of Inline Substitution: A Case Study.* ACM Letters on Programming Languages and Systems, 1(1):22–31, 1992.

An Empirical Study of Precise Interprocedural Array Analysis - Hind, Burke, Carini.. (1994) (8 citations) (Correct)

....of a called subroutine is substituted for the call statement with appropriate changes made to the naming of the formal parameters. **While selectively performing inlining can be beneficial, the cost of an enlarged program renders it infeasible as a general solution to handling all subroutine calls [47, 16].** Thus, inlining is used as a complement, rather than an alternative, to interprocedural analysis. **The** relationship between inlining and our precise summary method is discussed in Section 3.1. **Traditionally**, to determine the side effects of a call statement, two prior analyses of the called ....

Keith D. Cooper, Mary W. Hall, and Linda Torczon. *Unexpected Side Effects of Inline Substitution: A Case Study.* ACM Letters on Programming Languages and Systems, 1(1):22–32, March 1992.

A Study and Analysis of Function Inlining - Deshpande, Somani (Correct)

....If the register allocator decides to spill a heavily used variable which was previously in a register, the number of load stores would increase significantly. **Depending on the nuances of a particular compiler, there might be other side effects which affect the performance of the program [2].** In the rest of the report, we study the effects of programs inlining on program execution behavior and optimizer effectiveness. 2 Benchmark Programs We have measured 10 programs written in C and C . The C programs include five out of the six 1 programs from the SPECint92 benchmark suite, and ....

Cooper Keith D., Hall Mary W., and Torczon Linda, *Unexpected Side Effects of Inline Substitution: A Case Study*, ACM Letters on Programming Languages and Systems, March 1992, pp. 23-32.

Quantifying Behavioral Differences Between C and C++ Programs - Calder, Grunwald, Zorn (1995) (47 citations) (Correct)

....several challenges and opportunities for compiler writers and architects. **First, procedure inlining may be more promising, but more difficult to accomplish, due to the larger number of indirect function calls and the difficultly of interprocedural data flow analysis in C [35] Other studies [42, 43, 44] have shown that procedure inlining is problematic; it does not always improve program performance.** However, one attribute of procedure inlining is that it removes procedure calling overhead; this overhead is obviously a larger percentage of small procedures. **Thus**, automated procedure inlining ....

K. D. Cooper, M. W. Hall, and L. Torczon. *Unexpecledside effects of inline substitution: a case study.* Letters on Programming Languages and Systems, 1(1):22–32, March 1992.

Flow-directed Inlining - Jagannathan, Wright (1996) (40 citations) (Correct)

....While call string based analyses are highly precise, they also appear to be quite expensive to compute [15] In general, polymorphic splitting provides as much, if not more, accuracy than polyvariant call string analyses, but at significantly lower cost. 5. **2 Inlining Cooper, Hall, and Torczon [8, 9] studied the effects of inlining for Fortran programs.** They determined inlining sites by hand, and were able to eliminate the majority of dynamically executed procedure calls from their benchmark programs. **Furthermore**, they found that the opportunities revealed by inlining mitigated object code ....

Cooper, K., Hall, M., and Torczon, L. *Unexpected Side Effects of Inline Substitution: A Case Study*. ACM Letters on Programming Languages and Systems 1, 1 (1992), 22--32.

Interprocedural Analysis for Loop Scheduling and Data Allocation - Trung Nguyen (1998) (Correct)

....subroutine Foo. **In** this example, A and B map to X and Y respectively at the first call statement and to C and D respectively at the second call statement. **Hence**, the memory location represented by a formal parameter may vary from one invocation of the subroutine to another. **Subroutine in lining [7, 8, 11] can eliminate all formal parameters, and subroutine cloning [6] can result in a unique virtual address for every formal parameter.** However, in lining and cloning may make the resulting object code overly large and, thus, are performed to limited extent in practice. **When** we determine the ....

K. Cooper, M. Hall, and L. Torczon. *Unexpected side effects of inline substitution - a case study*. Letters on Programming Languages and Systems, 1(1), March 1992.

Inline Expansion: When and How? - Serrano (1997) (Correct)

....This is done by a combination of an inlining decision algorithm (deciding when to perform the expansions) and ad hoc expansion frameworks (describing how to perform the expansions) Inlining may impair code production because of los of high level informations. **For instance, Cooper et al. report in [5] that inlining discards aliasing informations in Fortran programs, so that compilers are unable to avoid interlock during executions.** Davison and Holler show in [6] that inlining for C may increase register save and restore operations because of C compilers artifacts. **On** our part, we have never ....

K. Cooper, M. Hall, and L. Torczon. *Unexpected Side Effects of Inline Substitution: A Case Study*. ACM Letters on Programming Languages and Systems, 1(1):22--31, 1992.

Simple and Effective Link-Time Optimization of Modula-3 Programs - Fernandez (1994) (50 citations) (Correct)

....inlining increases dependencies between source modules, and library procedures whose source is unavailable cannot be inlined. **Choosing call sites heuristically also has disadvantages: aggressive heuristics may trigger inlining at numerous sites and increase compilation time significantly [7]; the number of local variables in a caller can increase significantly, which increases register pressure and spilling and hence execution time [7, 14] and the heuristics are usually applied before the entire program is available, without knowing the execution frequencies of the inlined sites.** ....

.... **sites heuristically also has disadvantages: aggressive heuristics may trigger inlining at numerous sites and increase compilation time significantly [7] the number of local variables in a caller can increase significantly, which increases register pressure and spilling and hence execution time [7, 14]; and the heuristics are usually applied before the entire program is available, without knowing the execution frequencies of the inlined sites.** Profile guided inliners exist [6] but are used at compile time and thus suffer the same problems as source to source inlining. **Converting** method ....

K. D. Cooper, M. W. Hall, and L. Torczon. *Unexpected side effects of inline substitution: A case study*. ACM Letters on Programming Languages and Systems, 1(1):22--32, 1992.

Quantifying the Effects of Communication Optimizations - Choi (1997) (Correct)

....of a loop that access non local data, hence exposing additional computation that may be overlapped with communication. **Another** simple method that may increase the opportunities for optimization is to use procedure inlining. **Cooper et al. studied inlining in the context of scientific applications[8].** Though they found that it was almost always detrimental to performance, the presence of communication was not considered. **In** addition, we are currently investigating the interaction of communication optimizations with other array language optimizations, such as array contraction, and more ....

Keith D. Cooper, Mary W. Hall, and Linda Torczon. *Unexpected side effects of inline substitution: A case study*. ACM Letters on Programming Languages and Systems, 1(1):22--32, March 1992.

Link-Time Optimization of Modula-3 Programs - Fernandez, Hanson (Correct)

....the same module, increases dependencies between source modules, and library procedures whose source is unavailable cannot be inlined. **Choosing call sites heuristically also has disadvantages: aggressive**

heuristics may trigger inlining at numerous sites and increase compilation time significantly [6]; the number of local variables in a caller can increase significantly, which increases register pressure and spilling and hence execution time [6, 14] and the heuristics are usually applied before the entire program is available without knowing the execution frequencies of the inlined sites. mld ....

.... **sites heuristically also has disadvantages: aggressive heuristics may trigger inlining at numerous sites and increase compilation time significantly [6] the number of local variables in a caller can increase significantly, which increases register pressure and spilling and hence execution time [6, 14]; and the heuristics are usually applied before the entire program is available without knowing the execution frequencies of the inlined sites.** mld applies inlining at link time and chooses candidate sites using information gathered from profiles of the program s execution. **Link** time inlining ....

K. D. Cooper, M. W. Hall, and L. Torczon. *Unexpected side effects of inline substitution: A case study*. ACM Letters on Programming Languages and Systems, 1(1):22--32, 1992.

FIAT: A Framework for Interprocedural Analysis and.. - Carle, Hall.. (1995) (2 citations) **Self-citation (Hall)** (Correct)

....have not yet been implemented. **It** is not clear whether ALIAS information should be used when combining interproceduralinformation with dependence analysis. **Because ALIAS analysis is flow insensitive, it will falsely report aliases which may lead to overly conservative dependence information [20].** Since it is a violation of the Fortran standard for two variables to be aliased across a procedure call if one of the variables is modified, ignoring aliases is safe for all standard conforming Fortran programs. 17 Data race detection. **To** automatically detect data races during execution of a ....

K. D. Cooper, M. Hall, and L. Torczon. *Unexpected side effects of inline substitution - a case study*. Letters on Programming Languages and Systems, 1(1), Mar. 1992.

The ParaScope Parallel Programming Environment - Cooper (1993) (39 citations) **Self-citation (Cooper Hall Torczon)** (Correct)

....parallelism and improve the runtime behavior of the program [16] Unfortunately, these interprocedural techniques can have negative side effects. **ffl A study by Cooper, Hall, and Torczon using commercial Fortran compilers showed that inline substitution often resulted in slower running code [43, 44].** When this happened, secondary effects in the optimizer outweighed any improvements from inlining. **Inline** substitution introduces recompilation dependences and increases average procedure size. **It** can increase the overall source code size and compilation time. **ffl** Loop embedding and extraction ....

K. D. Cooper, M. W. Hall, and L. Torczon, "*Unexpected side effects of inline substitution: a case study*," to appear in Letters on Programming Languages and Systems, Mar. 1992.

FIAT: A Framework for Interprocedural Analysis and.. - Carle, Hall.. (1995) (2 citations) **Self-citation (Hall)** (Correct)

....have not yet been implemented. 5 It is not clear whether Alias information should be used when combining interprocedural information with dependence analysis. **Because Alias analysis is flow insensitive, it will falsely report aliases which may lead to overly conservative dependence information [20].** Since it is a violation of the Fortran standard for two variables to be aliased across a procedure call if one of the variables is modified, ignoring aliases is safe for all standard conforming Fortran programs. **Data** race detection. **To** automatically detect data races during execution of a ....

K. D. Cooper, M. Hall, and L. Torczon. *Unexpected side effects of inline substitution -- a case study*. Letters on Programming Languages and Systems, 1(1), Mar. 1992.

Online articles have much greater impact More about CiteSeer.IST Add search form to your site Submit documents Feedback

CiteSeer.IST - Copyright Penn State and NEC